IN THE CLAIMS

Please cancel Claims 3-4, 7-8 and 17-18.

Please amend Claims 1, 5, 12, 21 and 24 as follows:

1. (Currently Amended) A memory system, comprising:

a memory cache responsive to a single processing unit, the memory cache arrangeable to comprise:

a first independently cached area assigned to store a first number of data packets based on a first processing unit context; and

a second independently cached area assigned to store a second number of data packets based on a second processing unit context; and

a memory control system coupled to the memory cache, the memory control system configured to arrange the first independently cached area and the second independently cached area in such a manner that the first number of data packets and the second number of data packets coexist in the memory cache and are available for transfer between the memory cache and the single processing unit,

wherein the single processing unit comprises a central processing unit (CPU),

wherein the first processing unit context comprises a first CPU process and the second processing unit context comprises a second CPU process, and

wherein the first independently cached area is configurable in such a manner that at least some of the first number of data packets are transferable from the first independently cached area in response to execution of the first CPU process by the single processing unit, and wherein the second independently cached area is configurable in such a manner that at least some of the second number of data packets are transferable from the second independently cached area in response to execution of the second CPU process by the single processing unit.

2. (Original) The memory system according to claim 1, wherein the first number of data packets and the second number of data packets are concurrently available for bidirectional transfer between the memory cache and the single processing unit.

3-4. (Canceled)

5. (Currently Amended) The memory system according to claim ~~3~~ 1, wherein each data packet comprises a smallest unit of transaction handled by the memory cache.

6. (Original) The memory system according to claim 5, wherein each data packet comprises one of: a data word; a cache line and a number of bytes.

7-8. (Canceled)

9. (Original) The memory system according to claim 1, wherein data packets are transferable between the memory cache and the single processing unit based on requests from the single processing unit to the memory control system.

10. (Original) The memory system according to claim 1, wherein the memory control system comprises:

a first control register configured for association with the first independently cached area; and

a second control register configured for association with the second independently cached area,

the first and second control registers configured to store one of: a processing unit context assigned to the associated independently cached area; a memory location of the associated independently cached area in the memory cache; and a size of the associated independently cached area in the memory cache.

11. (Original) The memory system according to claim 10, wherein each of the first and second control registers is further configured to store a flag, the first number of data packets prohibited from entering the second independently cached area and the second number of data packets prohibited from entering the first independently cached area based on an assertion status of the flag.

12. (Currently Amended) A method, comprising:

providing a memory control system to control a memory cache responsive to a single processing unit;

allocating a first portion of the memory cache to store a first number of data packets associated with a first processing unit context, to form a first independently cached area;

allocating a second portion of the memory cache to store a second number of data packets associated with a second processing unit context, to form a second independently cached area;

arranging transfer of at least some of the first number of data packets between the memory cache and the single processing unit; and

arranging transfer of at least some of the second number of data packets between the memory cache and the single processing unit,

the first number of data packets and the second number of data packets being coexistent in the memory cache,

wherein data packets are arranged for transfer between the memory cache and the single processing unit based on requests from the single processing unit to the memory control system, and

further wherein transfer of the at least some of the first and second number of data packets in response to execution of the first and second processes, respectively, does not require the first and second processes to recognize allocation of the first independently cached area or the second independently cached area.

13. (Original) A computer-readable storage medium having stored thereon one or more software programs which, when executed, implement the method of claim 12.

14. (Original) The method according to claim 12, wherein the single processing unit comprises a central processing unit (CPU).

15. (Original) The method according to claim 14, wherein the first and second processing unit contexts comprise one of: processes; threads; and tasks.

16. (Original) The method according to claim 15, further comprising:

arranging transfer of at least some of the first number of data packets between the memory cache and the CPU in response to execution of a first process by the single processing unit; and

arranging transfer of least some of the second number of data packets between the memory cache and the CPU in response to execution of a second process by the single processing unit.

17-18. (Canceled)

19. (Original) The method according to claim 12, further comprising:

providing a first control register associated with the memory control system and a second control register associated with the memory control system, the first and second control registers associated with the first and second independently cached areas, respectively, and configured to store one of: a processing unit context type assigned to the associated independently cached area; a memory location of the associated independently cached area in the memory cache; and a size of the associated independently cached area in the memory cache.

20. (Original) The method according to claim 19, further comprising:

controlling assertion of a flag in each of the first and second control registers, the first number of data packets being prohibited from entering the second independently cached area and the second number of data packets from entering the first independently cached area based on an assertion status of the flag.

21. (Currently Amended) A computer system, comprising:

a bus;

a central processing unit coupled to the bus;

a system memory coupled to the central processing unit;

a memory cache coupled to the central processing unit, the memory cache arrangeable to comprise:

a first independently cached area assigned to store a first number of data packets associated with a first process executable by the central processing unit; and

a second independently cached area assigned to store a second number of data packets associated with a second process executable by the central processing unit; and

a memory control system coupled to the memory cache, the memory control system configured to arrange the first independently cached area and the second independently cached area in such a manner that the first number of data packets and the second number of data packets coexisting in the memory cache

wherein the first independently cached area is configurable in such a manner that at least some of the first number of data packets are transferable from the first independently cached area in response to execution of the first process by the central processing unit, and wherein the second independently cached area is configurable in such a manner that at least some of the second number of data packets are transferable from the second independently cached area in response to execution of the second process by the central processing unit.

22. (Original) The computer system according to claim 21, further comprising:

a plurality of instructions executable by the central processing unit to perform a method comprising:

alternately executing the first process and the second process, the first number of data packets and the second number of data packets coexisting in the memory cache and being concurrently available for transfer between the memory cache and the central processing unit.

23. (Original) The computer system according to claim 22, wherein the memory control system comprises:

a first control register associated with the first independently cached area; and

a second control register associated with the second independently cached area,

the first and second control registers configured to store one of: a process identifier assigned to the associated independently cached area; a memory location of the associated independently cached area in the memory cache; and a size of the associated independently cached area in the memory cache.

24. (Currently Amended) A processing unit, comprising:

a central processing engine operative to alternately execute a first process and a second process; and

a memory control interface to communicate with the central processing engine, the memory control interface operative to respond to a memory cache, the memory cache arrangeable to comprise:

a first independently cached area assigned to store a first number of data packets associated with the first process; and

a second independently cached area assigned to store a second number of data packets associated with the second process, the first number of data packets and the second number of data packets coexisting in the memory cache,

when alternately executing the first process and the second process, the central processing engine operative to cause the memory control interface to arrange for transfer of the first number of data packets and the second number of data packets, respectively, between the memory cache and the central processing engine, the first number of data packets and the second number of data packets being concurrently available for transfer

wherein the first independently cached area is configurable in such a manner that at least some of the first number of data packets are transferable from the first independently cached area in response to execution of the first process by the central processing engine, and wherein the second independently cached area is configurable in such a manner that at least some of the second number of data packets are transferable from the second independently cached area in response to execution of the second process by the central processing engine.